Traktorfuhrpark

Aufgaben

Die Digitalisierung hält schon seit vielen Jahren Einzug in die moderne Landwirtschaft. So sind Traktoren heutzutage oft vernetzte Fahrzeuge, die mittels GPS-Navigation automatisch die Spur halten können und Informationen über die eingesetzte Saatgut- und Düngemittelmenge sammeln und diese an eine Zentrale zur Auswertung weiterleiten. Solche Traktoren bedeuten eine hohe Investition für landwirtschaftliche Betriebe, so dass sich viele Betriebe die Traktoren bei Fuhrparks mieten.

1 Ziel ist es, einen Teil einer Online-Verwaltungssoftware für den Fuhrpark zu entwickeln. Kundinnen und Kunden (im Folgenden Kunden genannt) des Fuhrparks benötigen zunächst Registrierungsdaten, bestehend aus Username und Passwort, die sie von einer Sachbearbeiterin oder einem Sachbearbeiter (im Folgenden Sachbearbeiter genannt) des Fuhrparks erhalten. Bei dem Registrierungsvorgang werden auch Anschriftsdaten und eine Abbuchungserlaubnis für ein Bankkonto mit IBAN erfasst.
Um eine Buchung durchzuführen meldet sich der Kunde auf der Webseite an. Er wählt auf der Webseite einen Traktor nach Hersteller und Leistungsklasse aus und entscheidet, ob er einen Anhänger hinzubuchen möchte. Zur Auswahl stehen Transportanhänger und Saatgutverteiler. Die Transportanhänger sind nach unterschiedlichem Zuladegewicht auswählbar, die Saatgutverteiler nach der Saatkapazität in Quadratmetern pro Stunde. Jeder Traktor und jeder Anhänger haben einen Mietpreis pro Tag.
Als weiterer Service wird auch Saatgut mit einem Preis pro Zentner verkauft, welches einem Buchungsvorgang hinzugefügt werden kann. Die Verwaltungssoftware ermittelt aus den ausgewählten Traktoren und Anhängern sowie den zusätzlich gewünschten Saatgutmengen einen Angebotspreis. Der Kunde wählt einen freien Termin und kann den Angebotspreis akzeptieren und damit den Termin buchen.

1.1 Entwickeln Sie unter Berücksichtigung der obigen Beschreibung ein aussagekräftiges Use-Case-Diagramm und bestimmen Sie die Akteure, Beziehungen zwischen Akteuren und Anwendungsfällen und eventuell vorhandene include- bzw. extend-Beziehungen.

(10 BE)

1.2 Das UML-Klassendiagramm in Material 1 zeigt einen Ausschnitt aus der Verwaltungssoftware. Beschreiben und erläutern Sie das dargestellte Klassendiagramm unter Verwendung folgender Stichworte: Klasse, Referenzvariablen (Objektreferenzen), Operation (Methode), Datentyp, Sichtbarkeit, Vererbung, abstrakte Klasse, Parameter, Assoziation und Multiplizität.

Hinweis:
Die Attribute vom Datentyp int, double und String sowie alle Konstruktoren sind nicht zu beschreiben.

(10 BE)

1.3 Der Kunde Heimeler hat für den 27.7.2021 einen Traktor des Herstellers Schwerlast mit einer Leistung von 700 kW sowie einen Saatgutverteiler mit einer Saatkapazität von 6000 Quadratmetern pro Stunde für zwei Tage bestellt. Als Saatgut wurden 20 Zentner Saatgut

hinzugefügt. Die Angebotspreise betragen für den Traktor € 450,-/Tag, für den Saatgutverteiler € 180,-/Tag und für den Zentner Saatgut € 130,-.
Entwickeln Sie ein Objektdiagramm für das angegebene Szenario. Ergänzen Sie fehlende Informationen durch sinnvolle Werte.

(6 BE)

1.4 Implementieren Sie die Klassen `Buchung`, `Ware`, `Saatgut` und `Traktor` in einer in Ihrem Unterricht verwendeten objektorientierten Programmiersprache.

Hinweis: Die Operationen `myKosten()` und `angebotBerechnen()` sind nicht zu implementieren.

(9 BE)

1.5 Die Operation `angebotBerechnen()` in der Klasse `Buchung` berechnet den Angebotspreis für eine Buchung. Hierbei werden die Dauer der geplanten Miete in Tagen und die Kosten pro Tag sowie die Menge des bestellten Saatguts in Zentnern und die Kosten pro Zentner berücksichtigt. Implementieren Sie die Operation `angebotBerechnen()` in der Klasse Buchung und die Operationen `myKosten()` in den Klassen `Ware`, `Traktor` und `Saatgut`.

(5 BE)

1.6 Erläutern Sie den Begriff Polymorphie anhand des Klassendiagramms in Material 1.

(5 BE)

1.7 In dem Klassendiagramm in Material 1 gibt es Optimierungspotenzial hinsichtlich der Vererbung. Entwickeln und erläutern Sie einen Änderungsvorschlag.

(5 BE)

2 In Material 2 ist ein Saatgutbehälter dargestellt. Der Füllstand des Saatguts im Saatgutbehälter wird überwacht. Für die Überwachung wird ein Niveausensor „N“ eingesetzt, welcher eine „1“ liefert, wenn er mit Saatgut bedeckt ist. Zusätzlich wird ein Durchflusssensor „D“ verwendet, welcher überprüft, ob Saatgut durch die Förderorgane fließt. Ist Durchfluss vorhanden, liefert dieser eine „1“.
Außerdem wird die Drehung des Särads, das sich im Dosierorgan befindet, überwacht. Hierfür wird geprüft, ob das Särad eine Geschwindigkeit „V“ hat, wenn es eingeschaltet wird. Die Betätigung des Ein-Schalters „E“ liefert eine „1“. Dreht sich daraufhin das Särad, so liefert das Geschwindigkeitssignal „V“ eine „1“.

2.1 In Material 3 ist eine Wahrheitstabelle für sämtliche Sensorkombinationen abgebildet. Eine Störleuchte „S“ soll anzeigen, wenn der Saatgutbehälter leer ist (N = 0) oder wenn sich das Särad trotz Einschaltens nicht dreht (E=1 und V=0). Außerdem handelt es sich ebenfalls um eine Störung, wenn sich das Särad dreht, obwohl es nicht eingeschaltet ist (E=0 und V=1). Zusätzlich ist die Funktion gestört, wenn trotz Einschaltens (E=1) kein Durchfluss (D=0) vorhanden ist oder umgekehrt, wenn Durchfluss (D=1) ohne Einschalten (E=0) auftritt. Ergänzen Sie die Wahrheitstabelle.

(2 BE)

2.2 Ermitteln Sie die Funktionsgleichung für das Störsignal S in konjunktiver und disjunktiver Normalform.

Hinweis: Bei der Niederschrift der disjunktiven Normalform kann auf die Notation der UND-Zeichen verzichtet werden.

(6 BE)

2.3 Zeigen Sie mithilfe des KV-Diagramms in Material 4 auf, dass die vereinfachte disjunktive Gleichung $S = \bar{N} \vee (\bar{D} \wedge E) \vee (\bar{E} \wedge V) \vee (D \wedge \bar{V})$ gültig ist.

(4 BE)

2.4 Ermitteln Sie die Gleichung für NAND-Technik zu der Gleichung aus Aufgabe 2.3 und zeichnen Sie die Schaltung mit Logiksymbolen.

(6 BE)

2.5 Zur allgemeinen Betrachtung der Realisierung einer NAND-Schaltung mit integrierten Bauelementen wird die NAND-Schaltung $X = \overline{A \wedge B \wedge C \wedge D}$ zugrunde gelegt. Für die Umsetzung stehen integrierte Bauelemente vom Typ 74LS00 zur Verfügung (Material 5). Leiten Sie für X eine Gleichung her, die nur NAND-Verknüpfungen mit jeweils zwei Elementen enthält und zeichnen Sie den Verdrahtungsplan in Material 6.

(6 BE)

3 Die Steuerungs- und Überwachungsorgane (Material 2) arbeiten über einen Mikrocontroller. Dieser verarbeitet ein Start/Stopp-Signal und das Störsignal aus Aufgabe 2. Je nachdem, welche Sorte Saatgut auszubringen ist, ist die Dosierung einzustellen. Grobes Saatgut benötigt eine weitere Öffnung des Schiebers als feines Saatgut. Zudem ist die Geschwindigkeit des Särads einstellbar, um in Kurven nicht zu viel Saatgut auszubringen. Ergänzend ist es erforderlich, das restliche Saatgut komplett ablassen zu können. Dafür kann die Bodenklappe des Saatgutbehälters geöffnet werden. Die Steuerung wird über die Ein- und Ausgänge des Mikrocontrollers vorgenommen (Material 7).
Zu Beginn der Traktorfahrt startet das Mikrocontrollerprogramm automatisch über die Spannungsversorgung. Der Funktionsablauf wird im Folgenden beschrieben:

– Solange keine Störung (Störung bei Eingangssignal LOW) vorliegt wird der Funktionsablauf fortgesetzt.
– Zuerst wird der Saatguttyp abgefragt (Grobes Saatgut entspricht HIGH, feines Saatgut entspricht LOW).
– Erst wenn der Bediener mit dem Start/Stopp-Schalter das Säen startet (Start bei HIGH), wird der Schieber geöffnet. Der Schieber wird über zwei Bits gesteuert.
  Die Bitkombination LOW-LOW sorgt für ein komplettes Schließen des Schiebers.
  Die Bitkombination LOW-HIGH sorgt für die Öffnung für feines Saatgut.
  Die Bitkombination HIGH-LOW sorgt für die Öffnung für grobes Saatgut.
  Die Bitkombination HIGH-HIGH sorgt für das vollständige Öffnen des Schiebers.
– Das Särad wird in Bewegung gesetzt. Das erfolgt, solange das Signal für die Kurvenfahrt LOW ist, mit dem Aufruf des Unterprogramms `SOW_STRAIGHT`.
– Bei einer Kurvenfahrt erhält der Mikrocontroller ein HIGH-Signal. Während der Kurvenfahrt wird die Geschwindigkeit des Särads gedrosselt. Das erfolgt über den Aufruf des Unterprogramms `SOW_CURVE`.

- Bei Stopp (Start/Stopp-Schalter LOW) wird der Schieber geschlossen (LOW-LOW) und das Särad angehalten. Das Anhalten erfolgt über die Ausgabe von vier LOW-Signalen an den entsprechenden Ausgangsbits.
- Der geschlossene Schieber wird durch einen Endlagenschalter zurückgemeldet.
- Nachdem der Motor des Särads gestoppt wurde und die Schieberendlage erreicht wurde, ist es möglich die Bodenklappe zu öffnen, um das Saatgut komplett abzulassen. Der Bediener schaltet „Saatgut ablassen" ein (HIGH). Daraufhin wird die Bodenklappe mit einem HIGH-Signal für 30s geöffnet. Das Unterprogramm `WAIT_30s` sorgt für eine Pause von 30s.

3.1 Implementieren Sie die vollständige Portinitialisierung.

(2 BE)

3.2 Entwickeln Sie ein Struktogramm, das die beschriebene Funktionalität abbildet.

Hinweis: Die Unterprogramme `SOW_STRAIGHT`, `SOW_CURVE` und `WAIT_30s` sind als gegeben vorauszusetzen.

(12 BE)

3.3 Erläutern Sie den Nutzen, eine Störung mit einem LOW-Signal zu melden.

(3 BE)

3.4 Das Unterprogramm `SOW_CURVE` steuert den Motor des Särads über eine Vier-Bit-Kombination an und ist durch das Struktogramm in Material 8 beschrieben.
Implementieren Sie das Unterprogramm `SOW_CURVE`.

Hinweis: Das Unterprogramm `WAIT_10ms` ist als gegeben vorauszusetzen.

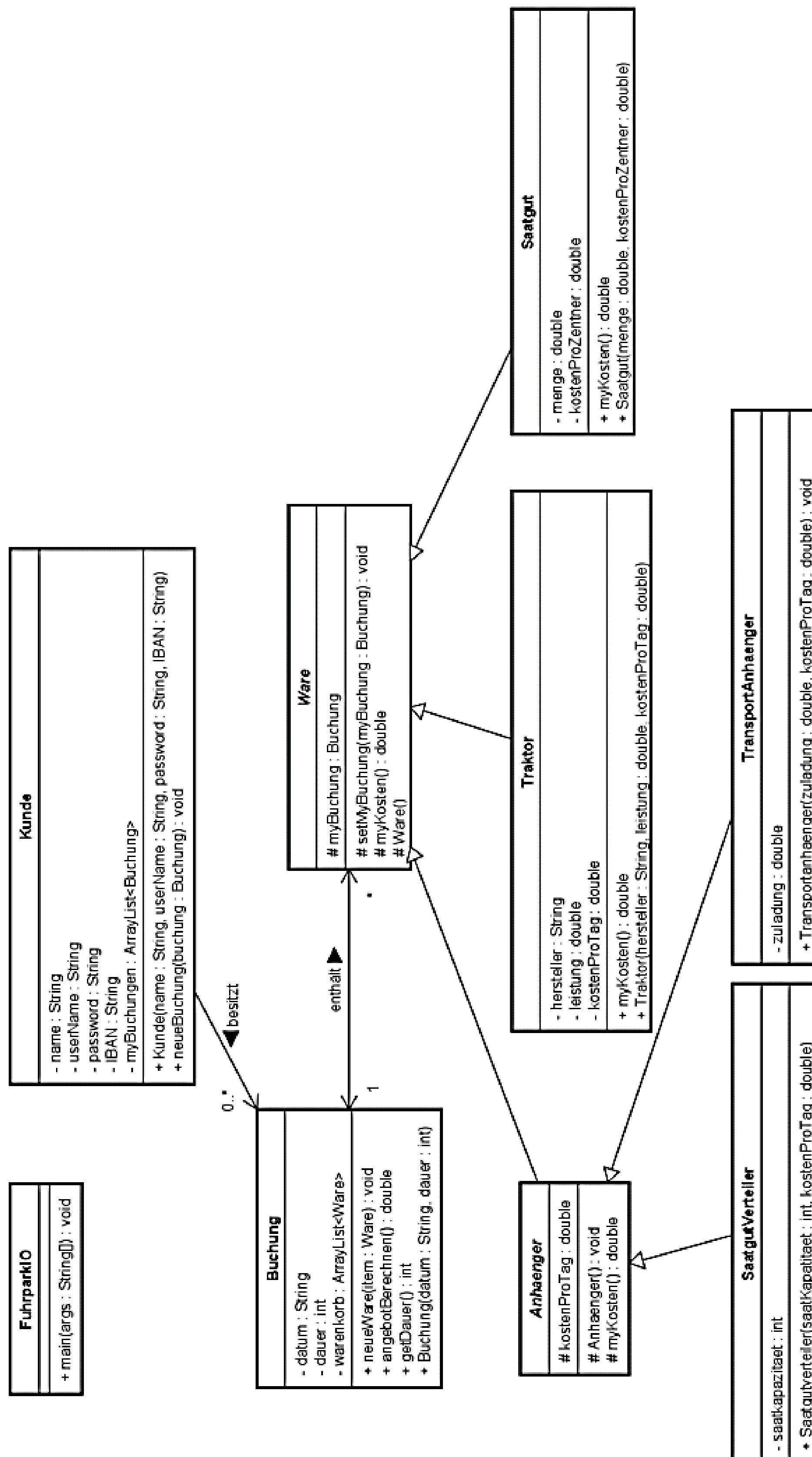
(6 BE)

3.5 Das Störsignal aus Aufgabe 2 wird u. a. mit einem Niveausensor ermittelt.
Erläutern Sie einen Sensortyp, mit dem die Erfassung des Füllstands des Saatguts im Saatgutbehälter umgesetzt werden kann.

(3 BE)

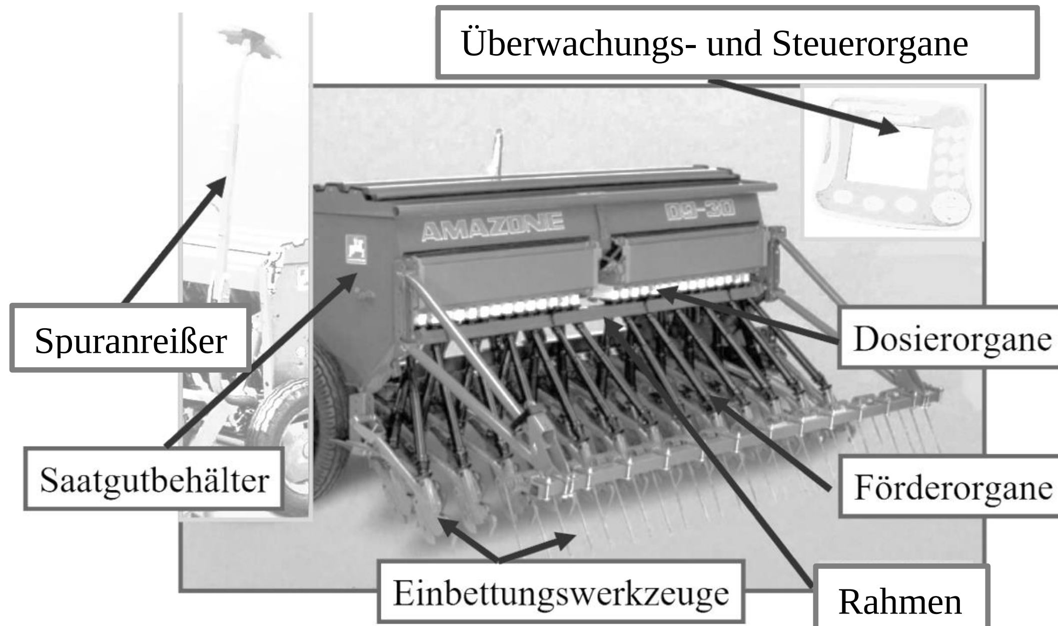
Material 1

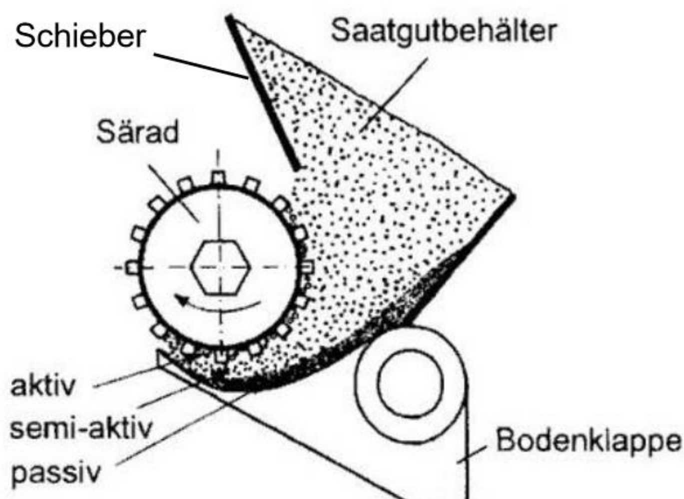
UML-Klassendiagramm



Material 2

Saatgutbehälter und Dosierorgane





https://tu-dresden.de/ing/maschinenwesen/int/ast/ressourcen/dateien/publikationen/Vorlesungsskripte/drillmaschinen.pdf?lang=de (abgerufen am 26.01.2021).

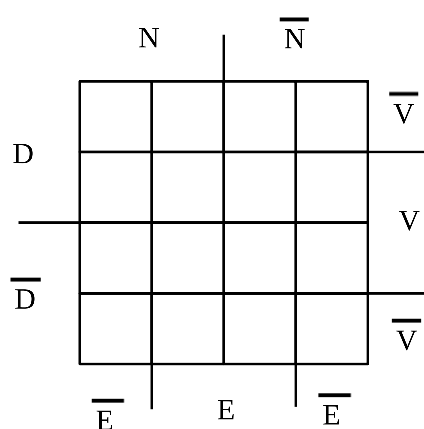
Material 3

Wahrheitstabelle

N	D	E	V	S
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

Material 4

KV-Diagramm für das Störsignal S



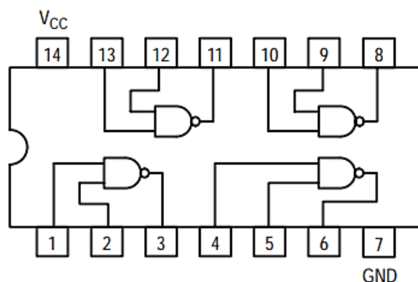
Material 5

Datenblatt des integrierten Bauelements vom Typ 74LS00

SN74LS00

Quad 2-Input NAND Gate

- ESD > 3500 Volts



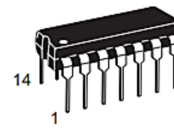


ON Semiconductor
Formerly a Division of Motorola
http://onsemi.com

LOW POWER SCHOTTKY

GUARANTEED OPERATING RANGES

Symbol	Parameter	Min	Typ	Max	Unit
V_{CC}	Supply Voltage	4.75	5.0	5.25	V
T_A	Operating Ambient Temperature Range	0	25	70	°C
I_{OH}	Output Current – High			–0.4	mA
I_{OL}	Output Current – Low			8.0	mA



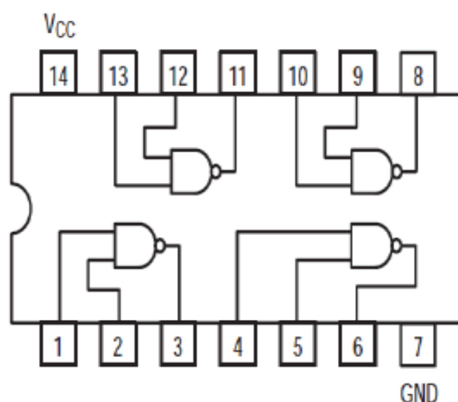
PLASTIC
N SUFFIX
CASE 646

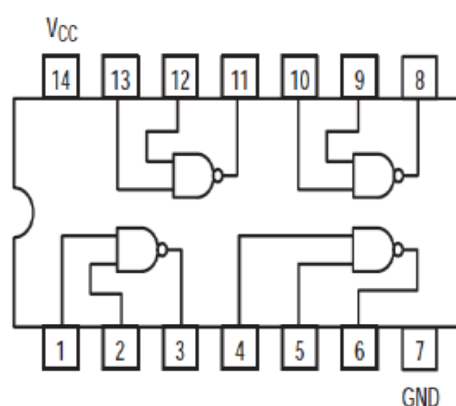
https://pdf1.alldatasheet.com/datasheet-pdf/view/12608/ONSEMI/7400.html (abgerufen am 25.01.2021).

Material 6

Verdrahtungsplan

A B C D X





Material 7

Portbelegung

Port 1 des Mikrocontrollers							
Pin 7	Pin 6	Pin 5	Pin 4	Pin 3	Pin 2	Pin 1	Pin 0
	↓	↓	↓	↓	↓	↓	↓
	Schieber Bit 1	Schieber Bit 0	Bodenklappe	Motorbit 3	Motorbit 2	Motorbit 1	Motorbit 0

Port 2 des Mikrocontrollers							
Pin 7	Pin 6	Pin 5	Pin 4	Pin 3	Pin 2	Pin 1	Pin 0
		↑	↑	↑	↑	↑	↑
		Störung	Start/Stopp	Saatguttyp	Kurvenfahrt	Saatgut ablassen	Endlage Schieber

Material 8

Struktogramm des Unterprogramms `SOW_CURVE`

```
Solange SOW_CURVE True
    Bit 0 und Bit 1 an / Bit 2 und Bit 3 aus
    WAIT_10ms
    Bit 1 und Bit 2 an / Bit 3 und Bit 0 aus
    WAIT_10ms
    Bit 2 und Bit 3 an / Bit 0 und Bit 1 aus
    WAIT_10ms
    Bit 3 und Bit 0 an / Bit 1 und Bit 2 aus
    WAIT_10ms
```